INFORMATION AND COMPUTATION 119, 154–159 (1995)

Loop Circuits and Their Relation to Razborov's Approximation Model

KATSUTOSHI NAKAYAMA

Semiconductor and Integrated Circuits Division, Hitachi, Ltd., Kodaira-shi, Tokyo 187, Japan

AND

AKIRA MARUOKA

Graduate School of Information Sciences, Tohoku University, Aoba-ku, Sendai 980, Japan

Recently, a new technique called the method of approximations has been developed for proving lower bounds on the size of circuits computing certain Boolean functions. To obtain a lower bound on the size complexity $size(f)$ of a certain function f by the method, an appropriate legitimate model $\mathcal{M}$ for the function f is chosen, and then a lower bound on the distance $\rho(f, \mathcal{M})$ from f to $\mathcal{M}$ is derived. The lower bound on $\rho(f, \mathcal{M})$ becomes a lower bound on $size(f)$ in view of the fact that $size(f) \geqslant \rho(f, \mathcal{M})$. Razborov gave a legitimate monotone model, $\mathcal{M}_{\text{mon}}(\mathcal{F}_{\max})$, and showed that $\rho(f, \mathcal{M}_{\text{mon}}(\mathcal{F}_{\max})) = \Omega(size^{1/3}(f))$, so there remains a gap between the size $size(f)$ and the distance $\rho(f, \mathcal{M})$. Employing his method, the following statements are established: (i) Razborov's model $\mathcal{M}_{\text{mon}}(\mathcal{F}_{\max})$ is generalized to obtain model $\mathcal{M}(\mathcal{F}_{\max})$, and it is established that $\rho(f, \mathcal{M}(\mathcal{F}_{\max})) = \Omega(size^{1/2}(f))$. (ii) Allowing the underlying graphs of circuits to have cycles, a new notion of apparently more powerful circuits, called loop circuits, is introduced, and it is proved that $\rho(f, \mathcal{M}(\mathcal{F}_{\max})) = \Theta(size_{\text{loop}}(f))$, where $size_{\text{loop}}(f)$ denotes the size complexity of f based on loop circuits.

1. INTRODUCTION

Recently, a number of strong results have been obtained to establish lower bounds for Boolean circuits and branching programs computing certain functions (Razborov, 1985a, 1991). Among the techniques developed for proving these bounds is the one called the method of approximations due to Razborov (1985a). Using the method exponential lower bounds have been obtained for monotone complexity (Andreev, 1985; Alon and Boppana, 1987; Razborov, 1985a, 1985b). Razborov (1989) studied the question of applying the method to arbitrary circuits and showed that a large number of auxiliary variables have to be introduced in constructing legitimate models in order to obtain strong lower bounds. In particular, it was proved that lower bounds which could be obtained by the method of approximations never exceed $O(n_0 n)$, where n_0 is the number of variables of a Boolean function in question and $n - n_0$ is the number of auxiliary variables introduced. To derive lower bounds on the size complexity $size(f)$ of function f, a legitimate model $\mathcal{M}$ is first determined and then a lower bound on the distance $\rho(f, \mathcal{M})$ is derived. The lower bound on $\rho(f, \mathcal{M})$ becomes a lower bound on $size(f)$ in view of the fact that $size(f) \geqslant \rho(f, \mathcal{M})$. Razborov gave a legitimate monotone model, $\mathcal{M}_{\text{mon}}(\mathcal{F}_{\max})$ and showed that $\rho(f, \mathcal{M}_{\text{mon}}(\mathcal{F}_{\max})) = \Omega(size^{1/3}(f))$, so there remains a gap between the size $size(f)$ and the distance $\rho(f, \mathcal{M})$. In Razborov's model, approximators (functions) in $\mathcal{M}$ are specified using monotone functions. In this paper, we remove the restriction of monotonicity and generalize Razborov's model $\mathcal{M}_{\text{mon}}(\mathcal{F}_{\max})$ to model $\mathcal{M}(\mathcal{F}_{\max})$. Based on the generalized model, it is established that $\rho(f, \mathcal{M}(\mathcal{F}_{\max})) = \Omega(size^{1/2}(f))$. Then, allowing the underlying graphs of circuits to have cycles, a new notion of apparently more powerful circuits, called loop circuits, is introduced and it is proved that $\rho(f, \mathcal{M}(\mathcal{F}_{\max})) = \Theta(size_{\text{loop}}(f))$, where $size_{\text{loop}}(f)$ denotes the size complexity of f based on loop circuits. This result gives an alternative characterization of the distance $\rho(f, \mathcal{M}(\mathcal{F}_{\max}))$ in terms of the size complexity based on loops circuits, which contrasts the characterization of the distance in terms of the minimal covering due to Razborov (1989).

2. PRELIMINARIES

For notation concerning approximations, we follow (Razborov, 1989). Let $\Sigma = \{0, 1\}$. For string w in Σ^n, let $w^{(i)}$ denote the ith symbol in w. For a in Σ and $1 \leqslant i \leqslant n$, let $X_a^{(i)}$ denote the set $\{w \in \Sigma^n \mid w^{(i)} = a\}$. Let B_n denote the set of Boolean functions of n variables. Let n variables of functions in B_n be denoted $x_1, x_2, \ldots, x_n$. Variable x_i and its negation $\neg x_i$ are also denoted x_i^1 and x_i^0, respectively. We sometimes regard variable x_i and its negation $\neg x_i$, together with constants 0 and 1, as functions in B_n in the obvious way.

0890-5401/95 $6.00

Moreover, $\vee$, $\wedge$, and $-$ are considered as functions from $B_n \times B_n$ to B_n as well: For any g and h in B_n, let

$$(g \vee h)(w) = g(w) \vee h(w),$$
$$(g \wedge h)(w) = g(w) \wedge h(w),$$
$$(g - h)(w) = g(w) \wedge \neg h(w).$$

A function f in B_n is considered as the set $\{w \in \Sigma^n \mid f(w) = 1\}$ as well. For g and h in B_n, let $g \leqslant h$ be defined as $g(w) \leqslant h(w)$ for any w in Σ^n, or equivalently $\{w \mid g(w) = 1\} \subseteq \{w \mid h(w) = 1\}$. When S is a set, let $|S|$ denote the number of elements in set S. Then $|f|$ denote the number of strings w's such that $f(w) = 1$. When C is a circuit, let $[\![C]\!]$ denote the function that C computes. The size complexity of f, denoted $size(f)$, is the minimum number of gates in circuits computing f.

An $\mathcal{M} \subseteq B_n$ closed under two binary operations $\bar{\vee}$ and $\bar{\wedge}$ is called a legitimate model when any variable x_i and its negation $\neg x_i$, and constants 0 and 1 belong to $\mathcal{M}$, where $1 \leqslant i \leqslant n$. If there exist a subset D_n of B_n and a bijective function φ from D_n to $\mathcal{M}$ such that

$$\varphi(x_i) = x_i,$$
$$\varphi(\neg x_i) = \neg x_i,$$
$$\varphi(0) = 0,$$
$$\varphi(1) = 1,$$
$$\varphi(g \vee h) = \varphi(g) \bar{\vee} \varphi(h),$$
$$\varphi(g \wedge h) = \varphi(g) \bar{\wedge} \varphi(h),$$

then we say $\langle D_n, \vee, \wedge \rangle$ is isomorphic to $\langle \mathcal{M}, \bar{\vee}, \bar{\wedge} \rangle$. Throughout the paper, $\varphi(f)$ is denoted $\bar{f}$.

Functions in $\mathcal{M}$ will be called approximators. We shall refer to $\bar{\vee}$ and $\bar{\wedge}$ as approximate OR and approximate AND, respectively. Given circuit C, let $\bar{C}$ denote the circuit obtained by replacing all the $\vee$ and $\wedge$ gates in C with $\bar{\vee}$ and $\bar{\wedge}$ gates, respectively. We shall call the circuit $\bar{C}$ the approximator circuit corresponding to C. Denote functions in $\mathcal{M}$ by $\bar{g}$, $\bar{h}$, or $\bar{f}$. Let

$$\delta_*^+(\bar{g}, \bar{h}) = \bar{g} * \bar{h} - \bar{g} \bar{*} \bar{h}, \tag{2.1}$$
$$\delta_*^-(\bar{g}, \bar{h}) = \bar{g} \bar{*} \bar{h} - \bar{g} * \bar{h}, \tag{2.2}$$

where $\bar{*} = \bar{\vee}$ when $* = \vee$, and similarly for $\wedge$. Put

$$\Delta^+ = \{\delta_*^+(\bar{g}, \bar{h}) \mid * \in \{\vee, \wedge\}, \bar{g}, \bar{h} \in \mathcal{M}\}, \tag{2.3}$$
$$\Delta^- = \{\delta_*^-(\bar{g}, \bar{h}) \mid * \in \{\vee, \wedge\}, \bar{g}, \bar{h} \in \mathcal{M}\}. \tag{2.4}$$

For f in B_n and $\bar{f}'$ in $\mathcal{M}$, let $\rho(f, \bar{f}')$ be the minimum s such that there exist $\bar{g}_1, \bar{h}_1, ..., \bar{g}_s, \bar{h}_s$ in $\mathcal{M}$ and $*_1, ..., *_s$ in $\{\vee, \wedge\}$ satisfying

$$f \leqslant \bar{f}' \vee \bigvee_{i=1}^{s} \delta_{*_i}^+(\bar{g}_i, \bar{h}_i), \tag{2.5}$$
$$\bar{f}' \leqslant f \vee \bigvee_{i=1}^{s} \delta_{*_i}^-(\bar{g}_i, \bar{h}_i). \tag{2.6}$$

Furthermore, define

$$\rho(f, \mathcal{M}) = \min_{\bar{f}' \in \mathcal{M}} \delta(f, \bar{f}'). \tag{2.7}$$

THEOREM 2.1 (Razborov, 1989). *Let $f = [\![C]\!]$, and $\bar{f} = [\![\bar{C}]\!]$. And let $*_i$ be the ith gate in C, and $\bar{g}_i$ and $\bar{h}_i$ be the input functions of the ith gate in $\bar{C}$. Then*

$$f \leqslant \bar{f}' \vee \bigvee_{i=1}^{s} \delta_{*_i}^+(\bar{g}_i, \bar{h}_i), \tag{2.8}$$
$$\bar{f}' \leqslant f \vee \bigvee_{i=1}^{s} \delta_{*_i}^-(\bar{g}_i, \bar{h}_i), \tag{2.9}$$

where s is the number of gates in C.

THEOREM 2.2 (Razborov, 1989). *For any Boolean function f and any legitimate model $\mathcal{M}$,*

$$\rho(f, \mathcal{M}) \leqslant size(f).$$

3. A GENERALIZATION OF RAZBOROV'S MODEL

From now on, let f in question is assumed to be given. In Razborov's model, denoted $\mathcal{M}_{\text{mon}}$, approximators are assumed to depend not only on variables on which function f depends, but also on auxiliary variables. Instead of adding auxiliary variables other than $x_1, ..., x_n$, functions f may be thought of as depending actually on variables $x_1, ..., x_{n_0}$, but not depending on remaining variables $x_{n_0+1}, ..., x_n$ which are considered to be auxiliary variables, where $n_0 \leqslant n$. Regarding f as a function of n_0 variables, we define subsets V and U of Σ^{n_0} as

$$V = f^{-1}(1),$$
$$U = f^{-1}(0).$$

For notational simplicity, f is considered as the function of n_0 variables as well as the corresponding one of n variables. Given v in V, let $\mathcal{F}_v$ denote the set of functions F from 2^U to Σ such that

$$F(\phi) = 0, \tag{3.1}$$
$$F(U) = 1, \tag{3.2}$$
$$F(U \cap X_a^{(i)}) = v^{(i)} \oplus a \oplus 1. \tag{3.3}$$

Note that in Razborov's model $\mathcal{M}_{\text{mon}}$ approximators F have to satisfy the condition of monotonicity in addition to the conditions mentioned above. Put

$$\mathcal{F}_{\max} = \bigcup_{v \in V} \mathcal{F}_v.$$

Let $\mathcal{F} \subseteq \mathcal{F}_{\max}$. Assuming for simplicity that $\log_2 |\mathcal{F}|$ is an integer, let

$$n = n_0 + \log |\mathcal{F}|,$$

and fix arbitrarily a one-to-one correspondence between Σ^{n-n_0} and $\mathcal{F}$. Later, we consider a function in $\mathcal{F}$ as the vector of length $n-n_0$ that corresponds to the function under the correspondence. Given any function g in B_{n_0}, the corresponding approximator, denoted $\bar{g}$, is a function in B_n specified as

$$\bar{g}(w, F) = \begin{cases} g(w), & \text{if } w \neq v(F), \\ F(U_g), & \text{otherwise,} \end{cases} \tag{3.4}$$

for w in Σ^{n_0} and F in $\mathcal{F}$. In (3.4), $v(F)$ is such that F is in $\mathcal{F}_{v(F)}$ and

$$U_g = \{u \in U \mid g(u) = 1\}.$$

Given $\mathcal{F} \subseteq \mathcal{F}_{\max}$ satisfying the conditions mentioned, $\mathcal{M}(\mathcal{F})$ is defined as

$$\mathcal{M}(\mathcal{F}) = \{\bar{g} \mid g \in \mathcal{F}\},$$

where $\bar{g}$ is given by (3.4). It is easy to see that $\bar{0} = 1$, $\bar{1} = 1$, and $\bar{x}_i^a = x_i^a$ hold for a in Σ. Approximate operations $\bar{\vee}$ and $\bar{\wedge}$ are defined so that $\langle B_{n_0}, \vee, \wedge \rangle$ is isomorphic to $\langle \mathcal{M}(\mathcal{F}_{\max}), \bar{\vee}, \bar{\wedge} \rangle$, namely, $\bar{g} \bar{\vee} \bar{h} = \overline{g \vee h}$ and $\bar{g} \bar{\wedge} \bar{h} = \overline{g \wedge h}$. Thus $\mathcal{M}(\mathcal{F})$ with approximate operations $\bar{\vee}$ and $\bar{\wedge}$ becomes a legitimate model. In particular, we refer to $\mathcal{M}(\mathcal{F}_{\max})$ as the maximal model.

LEMMA 3.1. *For any δ in $\Delta^+ \cup \Delta^-$ and any (w, F) in $\Sigma^{n_0} \times \mathcal{F}_{\max}$ such that $w \neq v(F)$,*

$$\delta(w, F) = 0.$$

LEMMA 3.2. *For any F in $\mathcal{F}_{\max}$ and $\bar{g}$ and any $\bar{h}$ in $\mathcal{M}(\mathcal{F}_{\max})$,*

$$\delta_{\vee}^{+}(\bar{g}, \bar{h})(v(F), F) = F(U_g) \vee F(U_h) - F(U_g \cup U_h),$$
$$\delta_{\wedge}^{+}(\bar{g}, \bar{h})(v(F), F) = F(U_g) \wedge F(U_h) - F(U_g \cap U_h).$$

Lemma 3.2 leads to the following notion of covering: For subsets A and B of U and F in $\mathcal{F}_{\max}$, $(A, B, \wedge)$ covers F if $F(A) \wedge F(B) = 1$ and $F(A \cap B) = 0$; $(A, B, \vee)$ covers F if $F(A) \vee F(B) = 1$ and $F(A \cup B) = 0$. A subset $\{(A, B, *)\}$ of $2^U \times 2^U \times \{\vee, \wedge\}$ covers $\mathcal{F}$ if for any F in $\mathcal{F}$ there exists $(A, B, *)$ in $\{(A, B, *)\}$ that covers F. Note that in Razborov's definition for covering, the latter case vanishes. This is because $F(A) \vee F(B) = 1$, together with the condition that F is monotone, implies $F(A \cup B) = 1$, which in turn implies that $(A, B, \vee)$ never covers F in the monotone case.

THEOREM 3.3. *$\rho(f, \mathcal{M}(\mathcal{F}))$ is equal to the minimum cardinality of a set of triples covering $\mathcal{F}$.*

Proof. Assume that there exist $\bar{f}_0$, $\bar{g}_1$, $\bar{h}_1$, ..., $\bar{g}_s$, $\bar{h}_s$ in $\mathcal{M}(\mathcal{F})$ and $*_1$, ..., $*_s$ in $\{\vee, \wedge\}$ such that

$$f \leqslant \bar{f}_0 \vee \bigvee_{i=1}^{s} \delta_{*_i}^{+}(\bar{g}_i, \bar{h}_i), \tag{3.5}$$

$$\bar{f}_0 \leqslant f \vee \bigvee_{i=1}^{s} \delta_{*_i}^{-}(\bar{g}_i, \bar{h}_i). \tag{3.6}$$

We shall construct $\{(A_1, B_1, *_1), ..., (A_s, B_s, *_s)\}$ that covers $\mathcal{F}$. The other direction of the proof will be omitted.

By Lemma 3.1, we have $\delta_{*_i}^{-}(\bar{g}_i, \bar{h}_i)(u, F) = 0$ for any (u, F) in $U \times \mathcal{F}$ and any $1 \leqslant i \leqslant s$. By (3.6) and (3.4), we therefore have $\bar{f}_0(u, F) = f_0(u, F) \leqslant f(u, F) = \bar{f}(u, F) = 0$ for any (u, F) in $U \times \mathcal{F}$, so we conclude that $f_0 \leqslant f$ and that $\bar{f}_0(u, F) = \bar{f}(u, F)$ for any (u, F) in $U \times \mathcal{F}$. Likewise, by Lemma 3.1 and (3.5), we have $1 = f(v, F) = \bar{f}(v, F) \leqslant \bar{f}_0(v, F)$, hence $\bar{f}(v, F) = \bar{f}_0(v, F)$ for any (v, F) in $V \times \mathcal{F}$ with $v \neq v(F)$. On the other hand, since $f_0 \leqslant f, \bar{f}_0(v(F), F) = F(U_{f_0}) = F(U_f) = F(\phi) = \bar{f}(v(F), F) = 0$ for any F in $\mathcal{F}$. Therefore we put $\bar{f} = \bar{f}_0$, so we have $f \leqslant \bar{f} \vee \bigvee_{i=1}^{s} \delta_{*_i}^{+}(\bar{g}_i, \bar{h}_i)$ by replacing $\bar{f}_0$ in (3.5) with $\bar{f}$, which implies

$$\bigvee_{i=1}^{s} \delta_{*_i}^{+}(\bar{g}_i, \bar{h}_i)(v(F), F) = 1$$

for any F in $\mathcal{F}$. Thus, putting $A_i = U \cap g_i^{-1}(1)$ and $B_i = U \cap h_i^{-1}(1)$ for $1 \leqslant i \leqslant s$, it is easy to see that $\{(A_i, B_i, *_i) \mid 1 \leqslant i \leqslant s\}$ covers $\mathcal{F}$ in view of Lemma 3.2. ∎

THEOREM 3.4. *Let f be a function in B_{n_0} such that $size(f) = \omega(n_0^2)$. Then for the maximal model $\mathcal{M}(\mathcal{F}_{\max})$ associated with f*

$$\rho(f, \mathcal{M}(\mathcal{F}_{\max})) = \Omega(size^{1/2}(f)).$$

Proof. Let $s = \rho(f, \mathcal{M}(\mathcal{F}_{\max}))$. By Theorem 3.3, there exists a set $\{(A_i, B_i, *_i) \mid 1 \leqslant i \leqslant s\}$ that covers $\mathcal{F}_{\max}$. As in the case of Razborov's model, it suffices to construct a circuit of the size $O((s+n_0)^2)$ to compute f. This is because $size(f) = O((s+n_0)^2)$, together with the condition $size(f) = \omega(n_0^2)$, implies $s = \omega(n_0)$, which in turn implies

$size(f) = O(s^2)$, and hence the statement of the theorem. Putting for $1 \leqslant i \leqslant n_0$

$$A_{s+i} = U \cap X_0^{(i)},$$
$$B_{s+i} = U \cap X_1^{(i)},$$
$$*_{s+i} = \wedge,$$

let S_0 denote

$$\{U\} \cup \{A_i \mid 1 \leqslant i \leqslant s+n_0\} \cup \{B_i \mid 1 \leqslant i \leqslant s+n_0\}$$
$$\cup \{A_i \cap B_i \mid *_i = \wedge\} \cup \{A_i \cup B_i \mid *_i = \vee\}.$$

Define the rules of inference on S_0:

$$A_i, B_i \vdash A_i \cap B_i, \qquad \text{if } *_i = \wedge, \tag{3.7}$$

$$\left.\begin{matrix} A_i \vdash A_i \cup B_i, \\ B_i \vdash A_i \cup B_i \end{matrix}\right\} \quad \text{if } *_i = \vee. \tag{3.8}$$

For $S \subseteq S_0$, let

$$\vdash(S) = \{Z \in S_0 \mid X, Y \in S, X, Y \vdash Z \text{ or } X \vdash Z\}.$$

Moreover, the closure of S is defined as

$$cl(S) = \bigcup_{i=1}^{\infty} \vdash^i (S),$$

where

$$\vdash^0 (S) = S,$$
$$\vdash^i (S) = \vdash(\vdash^{i-1}(S))$$

for $1 \geqslant 1$. Clearly

$$cl(S) = \bigcup_{i=1}^{|S_0|} \vdash^i (S).$$

The circuit to compute f is constructed based on the following fact.

FACT 3.5. *For w in Σ^{n_0}, $f(w) = 1$ if and only if $\phi \in cl(\{U\} \cup \{U \cap X_{w^{(i)}}^{(i)} \mid 1 \leqslant i \leqslant n_0\})$.*

Proof. Let S denote $cl(\{U\} \cup \{U \cap X_{w^{(i)}}^{(i)} \mid 1 \leqslant i \leqslant n_0\})$. For the proof of the "only if" part, assume in contradiction that $f(w) = 1$, i.e., $w \in V$, and $\phi \notin S$. Define function F, corresponding to S, from 2^U to Σ as $F(Y) = 1$ if there exists Y in S. Then $F(U \cap X_{w^{(i)} \oplus 1}^{(i)}) = 0$ for any $1 \leqslant i \leqslant n_0$. For, if $F(U \cap X_{w^{(i)} \oplus 1}^{(i)}) = 1$, then $U \cap X_{w^{(i)} \oplus 1}^{(i)} \in S$, together with $U \cap X_{w^{(i)}}^{(i)} \in S$, implies $\phi \in S$ by rule (3.7), which contradicts the assumption. The assumption also implies $F(\phi) = 0$. Moreover, we have $F(U) = 1$. Noting that $v(F)(= w)$ belongs to V, we therefore assert that F is in $\mathscr{F}_{\max}$. On the other hand, it is clear that no triple $(A_i, B_i, *_i)$ with $1 \leqslant i \leqslant s$ covers F, which is a contradiction.

For the proof of the "if" part, assume that $f(w) = 0$, i.e., $w \in U$. Then any member of $cl(\{U\} \cup \{U \cap X_{w^{(i)}}^{(i)} \mid 1 \leqslant i \leqslant n_0\})$ contains w, implying $\phi \notin cl(\{U\} \cup \{U \cap X_{w^{(i)}}^{(i)} \mid 1 \leqslant i \leqslant n_0\})$. ∎

In view of the fact, it is easy to construct the circuit computing f that is composed of $|S_0|$ rows, each consisting of $|S_0|$ gates. For D in S_0 and $1 \leqslant k \leqslant s$, let $f_{D,k}$ be the function that asserts that D can be deduced from $\{U\} \cup \{U \cap X_{w^{(i)}}^{(i)} \mid 1 \leqslant i \leqslant n_0\}$ within k steps. Then

$$f_{D,0} = \begin{cases} (x_i)^a, & \text{if } D = U \cap X_a^{(i)} \text{ for } a \text{ in } \{0, 1\} \\ 1, & \text{if } D = U, \\ 0, & \text{otherwise,} \end{cases}$$

$$f_{D,k+1} = f_{D,k} \vee \bigvee_{\substack{D = A_i \cap B_i \\ *_i = \wedge}} (f_{A_i,k} \wedge f_{B_i,k})$$
$$\vee \bigvee_{\substack{D = A_i \cup B_i \\ *_i = \vee}} (f_{A_i,k} \vee f_{B_i,k}).$$

Gates in the $(k-1)$th row are connected to those in the kth row according to the equality above. Then, by Fact 3.5, the gate in the $|S_0|$th row, corresponding to ϕ, computes f. Since $|S| = O(s + n_0)$, the theorem is established. ∎

The proof of Theorem 3.4 is similar to that of the corresponding theorem for the case of Razborov's monotone model. We notice here that Theorem 3.4 says that for Razborov's lower bound $size^{1/3}(f)$ on the distance ρ for the monotone model is replaced by $size^{1/2}(f)$ for the generalized model. The difference arises because the condition of monotonicity requires the rule of type $X \vdash Y$ for any X and Y in S_0 such that $X \subset Y$, and because in view of this rule the size of each row becomes $O(|S_0|^2)$, which implies that the size of the whole circuit is $O(|S_0|^3)$.

The argument in the proof of Theorem 3.4 works even if we replace $\mathscr{F}_{\max}$ in the statement of Theorem 3.4 with an appropriate smaller subset of $\mathscr{F}_{\max}$.

THEOREM 3.6. *Let f be a function in B_{n_0} such that $size(f) = \omega(n_0^2)$. Then there exists a subset $\mathscr{F}$ of $\mathscr{F}_{\max}$ such that*

$$\rho(f, \mathscr{M}(\mathscr{F})) \geqslant \Omega(size^{1/2}(f)),$$
$$|\mathscr{F}| \leqslant \exp(O(size(f)\,|U|)).$$

So the number of variables in $\mathscr{M}(\mathscr{F})$ is $O(size(f)\,|U|)$.

4. LOOP CIRCUIT

Loop circuits are defined in the same way as the usual Boolean circuits except that underlying graphs for loop

circuits can have cycles of dependency. A (Boolean) loop circuit consists of OR gates, AND gates, and input terminals labeled with variables $x_1, \neg x_1, ..., x_n, \neg x_n$, and constants 0 or 1 as in the case of usual Boolean circuits. Given an assignment for input variables, the values of gates in a loop circuit are determined as follows. At time 0 all of the values of gates are set to 0, while the values of input terminals are specified according to the assignment for input variables. Let g_i^j and h_i^j denote the values of two nodes connected to gate i at time j. Then the value of gate i at time $j+1$, denoted f_i^{j+1}, is determined by those of nodes at time j as $f_i^{j+1} = g_i^j *_i h_i^j$, where $*_i$ denotes the operation of gate i. Since gates in loop circuits are restricted to monotone ones, the collection of gates taking the value 1 increases monotonously as time goes on. So given an assignment to input variables for a loop circuit with s gates, gates in the circuit never change their values after time s. By using the values at time s, we can determine the function that a gate in a loop circuit computes in the obvious way. A loop circuit computes functions that gates in the circuit compute.) The circuit complexity, denoted $size_{\text{loop}}(f)$, based on loop circuits is defined in a similar way to the case of usual Boolean circuits: $size_{\text{loop}}(f)$ is defined to be the minimum number of gates in loop circuits computing f.

LEMMA 4.1. *Let f be a function in B_{n_0} such that $size_{\text{loop}}(f) = \omega(n_0)$. Then for the maximal model $\mathcal{M}(\mathcal{F}_{\max})$ associated with f*

$$\rho(f, \mathcal{M}(\mathcal{F}_{\max})) = \Omega(size_{\text{loop}}(f)).$$

Proof. Putting $s = \rho(f, \mathcal{M}(\mathcal{F}_{\max}))$, let $\{(A_i, B_i, *_i) \mid 1 \leqslant i \leqslant s\}$ denote a set of triples that covers $\mathcal{F}_{\max}$. As in the case of Theorem 3.6, it suffices to construct a loop circuit of size $O(s + n_0)$ to compute f. It is straightforward to construct the loop circuit whose unfolded version is identical with the circuit given in the proof of Theorem 3.4. ∎

LEMMA 4.2. *Let f be a function in B_{n_0}. For the maximal model $\mathcal{M}(\mathcal{F}_{\max})$,*

$$\rho(f, \mathcal{M}(\mathcal{F}_{\max})) \leqslant size_{\text{loop}}(f).$$

Proof. Let $s = size_{\text{loop}}(f)$. It suffices to show that there exist $\delta_1^+, ..., \delta_s^+$ in Δ^+ and $\delta_1^-, ..., \delta_s^-$ in Δ^- such that

$$f \leqslant \bar{f} \vee \bigvee_{i=1}^{s} \delta_i^+, \tag{4.1}$$

$$\bar{f} \leqslant f \vee \bigvee_{i=1}^{s} \delta_i^-, \tag{4.2}$$

where $\bar{f}$ is determined by f using (3.4). Since $\bar{f} \leqslant f$, (4.2) holds trivially no matter which error functions $\delta_1^-, ..., \delta_s^-$ in Δ^- we choose. So we shall specify $\delta_1^+, ..., \delta_s^+$ so that (4.1) is satisfied.

Let C_{loop} denote the circuit, with s gates, to compute f, let g_i and h_i denote the functions computed by the two gates preceding the ith gate in C_{loop}, and let f_i denote the function computed by the ith gate. Moreover, let $*_i$ denote the operation of the ith gate in C_{loop}. As in the case of usual Boolean circuits, let $\bar{C}_{\text{loop}}$ denote the circuit obtained by replacing all the $\vee$ and $\wedge$ gates in C_{loop} with $\bar{\vee}$ and $\bar{\wedge}$ gates, respectively. By nodes, we mean gates or input terminals in circuits. When g is computed at (or assigned to) a node in C_{loop}, assign $\bar{g}$ to the corresponding node in $\bar{C}_{\text{loop}}$, where $\bar{g}$ is determined by g using (3.4). In particular, $\bar{f}$ is assigned to the output gate in $\bar{C}_{\text{loop}}$ which corresponds to the output gate in C_{loop} computing f. Then since model $\mathcal{M}(\mathcal{F}_{\max})$ with approximate operations $\bar{\vee}$ and $\bar{\wedge}$ is isomorphic to B_{n_0} with Boolean operations $\vee$ and $\wedge$, we have $\bar{f}_i = \bar{g}_i \bar{*}_i \bar{h}_i$ for $1 \leqslant i \leqslant s$. Furthermore, the function associated with an input terminal in C_{loop} is identical to that associated with the corresponding input terminal in $\bar{C}_{\text{loop}}$. This is because $\bar{x}_i = x_i$, $\neg \bar{x}_i = \neg x_i$, $\bar{1} = 1$ and $\bar{0} = 0$ hold.

Using the approximators $\bar{g}_i$ and $\bar{h}_i$ above, put

$$\delta_i^+ = \delta_{*_i}^+(\bar{g}_i, \bar{h}_i)$$

for $1 \leqslant i \leqslant s$. We say that the ith node has positive error on v when $f_i(v) = 1$ and $\bar{f}_i(v) = 0$. For the proof of (4.1), assume that the output gate has positive error on v, i.e., $f(v) = 1$ and $\bar{f}(v) = 0$. Define set I of gates in $\bar{C}_{\text{loop}}$ recursively as follows: The output gate belongs to I; if the ith gate belongs to I and a gate preceding the ith gate has positive error on v, the preceding gate belongs to I. Let J denote the set consisting of nodes in $\bar{C}_{\text{loop}}$ that do not belong to I. Note that any gate in J that precedes a gate in I does not have positive error on v and that any input terminal in $\bar{C}_{\text{loop}}$ belongs to J. We may regard $\{I, J\}$ as the partition of nodes in $\bar{C}_{\text{loop}}$ as well as that of nodes in C_{loop}.

Assume in contradiction that $\delta_i^+(v) = \delta_{*_i}^+(\bar{g}_i, \bar{h}_i)(v) = 0$ for any $1 \leqslant i \leqslant s$. Moreover, suppose that there exists a gate, say the ith gate, in I such that both of its preceding gates belong to J. Then we have

$$f_i(v) = 1, \tag{4.3}$$

$$\bar{f}_i(v) = 0. \tag{4.4}$$

Furthermore, since the preceding gates do not have positive error on v, we have

$$g_i(v) \leqslant \bar{g}_i(v), \tag{4.5}$$

$$h_i(v) \leqslant \bar{h}_i(v). \tag{4.6}$$

It follows from (4.3), (4.4), (4.5), and (4.6) that

$$\begin{aligned} 1 &= f_i(v) \\ &= (g_i *_i h_i)(v) \\ &\leqslant (\bar{g}_i *_i \bar{h}_i)(v) \\ &= 1. \end{aligned}$$

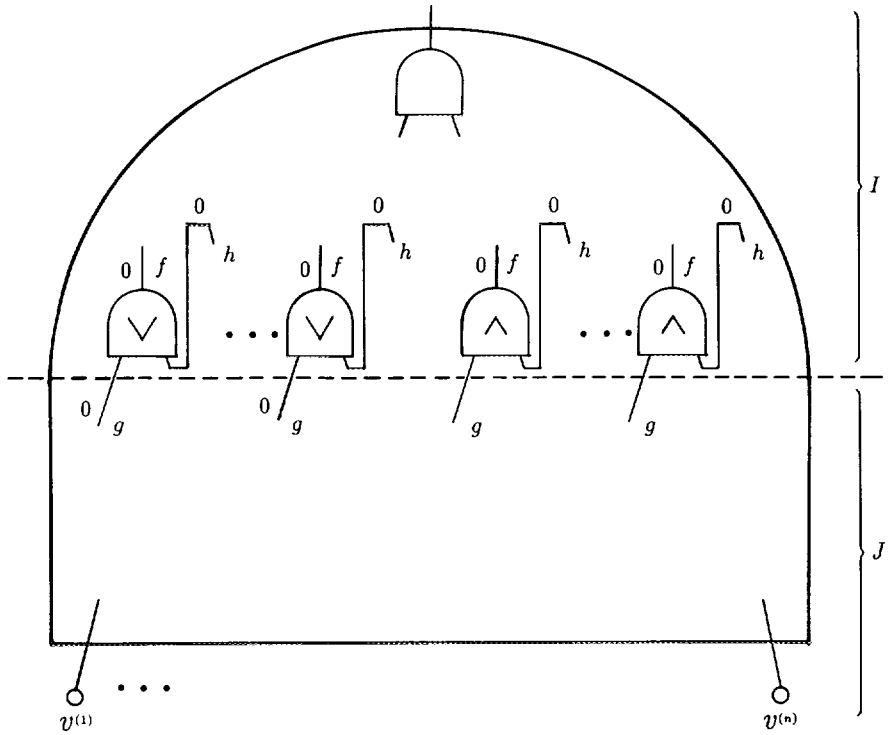

FIG. 1. Partition of the collection of gates to subsets I and J.

Thus, since

$$(\bar{g}_i \bar{*}_i \bar{h}_i)(v) = \bar{f}_i(v) = 0,$$

we have

$$\begin{aligned}\delta^+_{*_i}(\bar{g}_i, \bar{h}_i)(v) &= (\bar{g}_i *_i \bar{h}_i - \bar{g}_i \bar{*}_i \bar{h}_i)(v) \\ &= (\bar{g}_i *_i \bar{h}_i)(v) - (\bar{g}_i \bar{*}_i \bar{h}_i)(v) \\ &= 1 - 0 \\ &= 1,\end{aligned}$$

which is a contradiction. So we can assert that any gate in I preceded by a gate in J is also preceded by a gate in I. Let the ith gate be such a gate, and let g_i be computed by a gate in J and h_i be computed by a gate in I. Since $\delta_i^+(v) = (\bar{g}_i *_i \bar{h}_i)(v) - (\bar{g}_i \bar{*} \bar{h}_i)(v) = 0$ and $\bar{f}_i(v) = (\bar{g}_i \bar{*} \bar{h}_i)(v) = 0$, we have $(\bar{g}_i *_i \bar{h}_i)(v) = 0$. Suppose that $*_i = \vee$. Then $\bar{g}_i(v) = 0$ holds by $(\bar{g}_i \vee \bar{h}_i)(v) = \bar{g}_i(v) \vee \bar{h}_i(v) = 0$. We therefore have $g_i(v) = 0$, because $g_i(v) = 1$ implies that the gate computing g_i has positive error on v, hence it belongs to I, which is a contradiction. On the other hand, if $*_i = \wedge$, then we do not care about the value of $g_i(v)$. It is easy to see that, since any gate in I preceded by a gate in J is also preceded by a gate in I that takes as output value 0 on v at time 0, any gate in I adjacent to a gate in J takes as output value 0 on v at any time no matter what operation the gate has. See Fig. 1. This contradicts the fact that the output gate in C_{loop} takes as output value 1 on v at time s. ∎

By Lemma 4.1 and 4.2, we establish the following theorem.

THEOREM 4.3. *Let f be a function in B_{n_0}. For the maximal model $\mathcal{M}(\mathcal{F}_{\max})$,*

$$size_{\text{loop}}(f) \geq \rho(f, \mathcal{M}(\mathcal{F}_{\max})) = \Omega(size_{\text{loop}}(f)).$$

Received February 23, 1993; final manuscript received January 4, 1994

REFERENCES

Andreev, A. E. (1985), On one method of obtaining lower bounds of individual monotone function complexity, *Dokm. Akad. Nauk.* **282**, 1033–1037.

Alon, N., and Boppana, R. B. (1987), The monotone circuit complexity of Boolean functions, *Combinatorica* **7**(1), 1–22.

Razborov, A. A. (1985a), Lower bounds on the monotone complexity of some Boolean functions, *Dokm. Akad. Nauk.* **281**, 798–801.

Razborov, A. A. (1985b), A lower bound on the monotone network complexity of the logical permanent, *Math. Zametki* **37**(6), 887–900.

Razborov, A. A. (1989), On the method of approximations, *in* "Proceedings of 21st Annual ACM Symposium on Computing," pp. 167–176.

Razborov, A. A. (1991), Lower bounds for deterministic and nondeterministic branching programs, *in* "Proceedings 8th FCT," Lecture Notes in Computer Science, Vol. 529, pp. 47–60, Springer, New York/Berlin.